SIVP manuscript No.
(will be inserted by the editor)

Stereo Vision Based Road Estimation Assisted by Efficient Planar Patch Calculation

U. Ozgunalp · X. Ai · N. Dahnoun

Received: date / Accepted: date

Abstract A robust and accurate road model estimation algorithm can greatly improve the performance of many Advanced Driver Assistance Systems (ADAS) applications such as lane detection, obstacle detection and road marking recognition. To estimate the road model, the proposed algorithm employs a stereo vision camera system. In this paper, local planar patches are efficiently estimated in the disparity domain rather than conventionally in the Euclidean domain. Then, the estimated planar patch orientations are integrated to the fitting stage and orientation differences are minimized along with height differences. Moreover, patch orientation differences are exploited for weighting data points. Thus, outliers become insignificant in the fitting stage and the road model is estimated robustly and accurately without any prior knowledge of any extrinsic camera parameters. Experiments have been carried out for a free space calculation application and the road is segmented with a True Positive Rate (TPR) of 88%.

Keywords Road model estimation · ADAS · stereo vision · Disparity map · Quadratic fitting

1 Introduction

Estimation of the road surface is an essential component of many applications for intelligent vehicles such as lane detection, road marking recognition, free space computation and obstacle detection. An obstacle can be defined as any object which appears elevated from the road surface. Thus, the estimated road model can be used as a reference for obstacle detection.

For lane detection, traditionally, the vertical road profile (distance (z) vs height (y)) is estimated by using a single camera. Thus, to detect lanes, these algorithms need to use many assumptions for robust results, such as the known extrinsic camera parameters, a flat road and parallel lanes [22]. In [2], first, road sides (road boundaries or lane markings) are detected, then by assuming the road sides are parallel to each other, vertical (distance vs height) and horizontal (distance vs lateral offset) road profiles are estimated. In [4], the authors demonstrated a robust algorithm that filters estimated lanes and objects on the road (for instance the leading vehicle) using a single filter. Since, leading vehicles are likely to follow the lane, estimation of the lane geometry can be improved. In [14], information from the Internal Measurement Unit (IMU) and steering angle is also integrated. In [8], apart from lane markings and leading vehicles, road barriers are also integrated to the filtering. Despite the importance of the road slopes, in this algorithm, the road is assumed to be planar. This assumption could be tackled by a stereo vision based road surface estimation algorithm. More recent lane detection algorithms, use the 3D map acquired from active or passive sensors. Estimating the vertical road profile from a 3D map eliminates many assumptions such as the flat road and/or parallel lanes assumption and it is more robust than using a 2D map. Furthermore, it is possible to segment the feature map by segmenting the road points from the obstacles, based on their height.

The v-disparity map is a a common approach for the road surface modeling. The v-disparity map [11]

U. Ozgunalp, X. Ai and N. Dahnoun
Department of Electrical and Electronic Engineering, University of Bristol, Bristol, UK

U. Ozgunalp
E-mail: umar.ozgunalp@bristol.ac.uk
X. Ai
E-mail: ai.xiao@bristol.ac.uk
N. Dahnoun
E-mail: naim.dahnoun@bristol.ac.uk

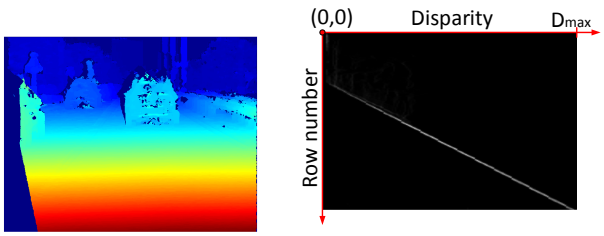

Fig. 1 An example disparity map is shown on the left and its estimated v-disparity map is shown on the right, where v-disparity map is created by accumulating the disparity values of the pixels for each row of the image.

is an accumulator with the axes of v (row number of the image) vs. d (disparity value of the pixel). The v-disparity map takes the disparity map as an input and accumulates each pixel's disparity value row by row (v). This approach assumes the roll angle is zero and there is no curvature in this axis. Thus, in each row, the disparity values of the pixels on the road should have the same disparity value. When moving the upper rows of the image, the distance of the road gradually increases (the disparity value is inversely proportional to the distance). Consequently, the road forms a line in the v-disparity map as can be seen in Figure 1. In Figure 1, on the left the input disparity map is shown, and on the right its v-disparity map is shown. For the flat road case (such as one in Figure 1), the disparity value of the road at the horizon line is zero and its disparity value increases linearly while going through the lower rows of the disparity map. Thus, the road can be approximated as a linear line in the v-disparity map (assuming the roll angle of the camera is zero). For this reason, in [7], the corresponding ground plane is modeled as a linear line and its parameters are extracted by using the Hough transform.

More robust algorithms require the extraction of the road model accurately even for non-flat roads. In [12], the authors assume that the road can be modeled as piecewise linear and non-flat roads can be modeled by a few linear lines (typically 3). In the optimization stage, this paper also uses the Hough transform and estimates the parameters of these linear lines. More recently, many algorithms prefer to use the Euclidian space rather than the disparity space. The Euclidian space preserves the physical properties of the road and is more suitable for the quadratic road models [19]. This is due to the nonlinear transformation from the WCS to the disparity space. A comparative study, published by [20], suggests that fitting in the Euclidian space is more accurate than applying the Hough transform to the v-disparity map. In [16], similar to the v-disparity map creation, the input 3D map (Euclidian space) is reduced to a 2D map by accumulating all the points into a histogram of height vs. distance. The algorithm, first, estimates the near-field with a linear model (using only the points from the near field) and, then, the road curvature (1 parameter) based on parameters detected from the near-field using only the points from the far field. For more flexibility, [23] uses a height vs distance histogram and models the road with a spline.

All of the methods discussed above assume the roll angle is zero and the road towards the axis, which is parallel to the roll angle, is flat. Thus, the estimation of the 2D road model is reduced to estimating a 1D model. To remove these assumptions and improve accuracy, [17] applies a 2D quadratic surface fitting. This model takes account of drainage gradients and is especially suitable for detecting curbs, potholes and small objects on the road. For increased accuracy, outliers are eliminated by checking the point densities on the density map and applying RANSAC based fitting. A similar road model (a 2D quadratic surface) is used in our previous paper [26]. In this model, twisting of the road surface (e.g. the left side of the road is higher in the near field but the right side is higher in the far field) can be represented by introducing one more parameter to the quadratic surface. This paper also introduces a method to estimate the planar patches for the Euclidian domain from the disparity map (working in the 2D space is much less computationally complex than working in the 3D space). Then, the estimated patch parameters (both the position and normal vectors) are exploited for eliminating outliers during road fitting. The system described in [26] extracts and utilizes valuable local information for global optimization and demonstrates robust results. However, it has the following drawbacks:

1. The system uses a 3D camera which is relatively sensitive to the illumination source

2. Initial approximate extrinsic camera parameters need to be known. Although these parameters do not need to be accurate, a practical system should not need such information as an input.

3. Outliers are thresholded and eliminated in a binary fashion. Even though, most of the outliers can be eliminated with the described method, all the remaining outliers contribute to the fitting stage as much as the inliers do. For more accurate results, each point should have its own weight

The algorithm that will be presented in this paper manages to eliminate most of the assumptions and solve the problems described above. In the proposed algorithm, planar patch normals are efficiently extracted from the disparity space and estimated patch normals are directly integrated to the fitting stage in the Euclidean space. Furthermore, robust fitting is achieved by

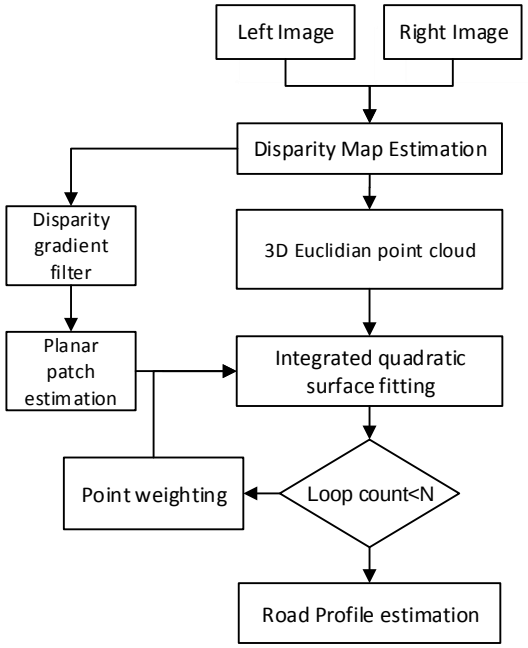

Fig. 2 Overview of the road model estimation algorithm.

iteratively fitting a quadratic road model to the point cloud data and calculating point weights using both estimated planar patch positions and normals. A block diagram of the system is illustrated in Figure 2.

2 PROBLEM FORMULATION

The proposed algorithm, estimates a quadratic road surface given the disparity map as an input. The used quadratic road surface model is illustrated in Figure. 3, and the road model used in this paper is defined as in Eqn. 1.

$$Y(x,z) = a + b \cdot x + c \cdot z + d \cdot x^2 + e \cdot z^2 + f \cdot x \cdot z \quad (1)$$

where $[a, b, c, d, e, f]$ are the parameters of the model, and $Y(x, z)$ is the predicted height of the road at location $[x, z]$. In this paper, planar patch orientations are efficiently estimated in the disparity domain and exploited in the fitting stage. Thus, the effects of the outliers, such as the points on the obstacles, are minimized.

3 DISPARITY MAP ESTIMATION

In this paper, the input depth map is estimated by using a stereo vision camera. Our stereo vision camera (see Section 7.1 for more details) supplies synchronized (in the time domain) and rectified images. When a point in a world coordinate system is captured with two different cameras, let $P(u_L, v_L)$ be the projection of the point on the left image, and let $P(u_R, v_R)$ be the projection of the point on the right image, where u is the horizontal coordinate of the pixel, and v is the vertical coordinate of the pixel. Then, in the rectified images, $v_L = v_R$, and $d = u_L - u_R.$, where d is the disparity between the corresponding points. When the disparity values of the pixels, instead of intensity values, are mapped to an image, then that image is called disparity map.

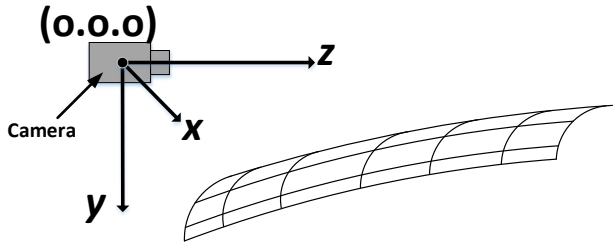

Fig. 3 An example quadratic road surface with respect to the camera coordinate system.

With known focal length (f), and baseline (b_s, the distance between left and right cameras) a point in disparity map $[u, v, d]$ can be transformed into Euclidean domain using Eqn. 2 [3].

$$z = \frac{f.b_s}{d}, \quad x = \frac{u.z}{f}, \quad y = \frac{v.z}{f} \quad (2)$$

There are many disparity map estimation algorithms in the literature [13]. However, the selected algorithm should be able to achieve good accuracy while working in real-time, such as [5] and [9]. The disparity map estimation algorithm used in this paper is our previously published algorithm [25]. The algorithm is selected due to its good accuracy and high computational efficiency.

4 PLANAR PATCH CALCULATION

In our previous system [26], a novel planar patch extraction algorithm was proposed, which avoids the computational complexities involved with traditional 3D algorithms [21], and it can be described as follows. A plane in the disparity domain is parametrized as in Eq. (3) where u (horizontal location), v (vertical location) and d (disparity) are the known parameters of a point in the disparity map and n_u, n_v, n_d and ρ_d are the parameters of the plane which needs to be estimated. By rearranging Eq. (3) as Eq. (4), partial derivatives with

respect to u and v can be estimated as Eq. (5) and Eq. (6).

$$n_u.u + n_v.v + n_d.d + \rho_d = 0 \tag{3}$$

$$d = -n_u/n_d.u - n_v/n_d.v - \rho_d/n_d \tag{4}$$

$$\delta d/\delta u = -n_u/n_d \tag{5}$$

$$\delta d/\delta v = -n_v/n_d \tag{6}$$

The plane equation in Eq. (3) can also be written in the form as in Eq. (7) by dividing by n_d. Replacing n_u/n_d with $-\delta d/\delta u$ and n_v/n_d with $-\delta d/\delta v$ in Eq. (7) yields Eq. (8), which represents the same plane.

$$n_u/n_d.u + n_v/n_d.v + d + \rho_d/n_d = 0 \tag{7}$$

$$-\delta d/\delta u.u - \delta d/\delta v.v + d + \rho_D = 0 \tag{8}$$

where $\rho_D = \rho_d/n_d$. By applying a gradient filter on the disparity image, $\delta d/\delta u$ and $\delta d/\delta v$ can be estimated and, then, by rearranging Eq. (8), ρ_D can be calculated as Eq. (9).

$$\rho_D = \delta d/\delta u.u + \delta d/\delta v.v - d \tag{9}$$

A point in the disparity domain $[u, v, d]$ can be transformed into the Euclidian domain $[x, y, z]$ using Eq. (10) [3], where f is the focal length and b_s is the base-line (the image origin is set to the image center). Thus, the plane equation shown in Eq. (8) can be rewritten as Eq. (11) by replacing $[u, v, d]$ using Eq. (10). Then by multiplying Eq. (11) with z and rearranging as Eq. (12) yields a similar form of the plane equation in the Euclidian space, as can be seen in Eq. (13).

$$u = f.\frac{x}{z}, \qquad v = f.\frac{y}{z}, \qquad d = f.\frac{b_s}{z} \tag{10}$$

$$-f.\delta d/\delta u.x/z - f.\delta d/\delta v.y/z + f.b_s/z + \rho_D = 0 \tag{11}$$

$$-f.\delta d/\delta u.x - f.\delta d/\delta v.y + \rho_D.z + f.b_s = 0 \tag{12}$$

When the plane equation is written in the form of Eq. (13), n_x, n_y and n_z can be calculated with the Eq. (14) which yields the parameters of the local planar patches.

$$n_x.x + n_y.y + n_z.z + \rho = 0 \tag{13}$$

$$n_x = -f.\delta d/\delta u, \quad n_y = -f.\delta d/\delta v, \quad n_z = \rho_D \tag{14}$$

Thus, knowing the intrinsic camera parameters, the parameters of the local planar patches can be estimated by simply applying a gradient filter to the disparity map.

5 INTEGRATED SURFACE FITTING

The first step of the quadratic surface fitting is to define the road model. The applied quadratic road model is defined as in Eqn. 1. Then, for a point i, predicted height by the road model is defined by Y_i (see Eqn. 15).

$$Y_i = a + b \cdot x_i + c \cdot z_i + d \cdot x_i^2 + e \cdot z_i^2 + f \cdot x_i \cdot z_i \tag{15}$$

where, $[x_i, y_i, z_i]$ is the location of the point in the point cloud, and $[a, b, c, d, e, f]$ are the parameters of the road surface. The objective of the fitting is to minimize the error. Traditionally, this error term (S) is defined as the sum of the squared differences (for the height difference). Thus, the S term can be defined as:

$$S = \sum_{i=1}^{n} (y_i - Y_i)^2 \tag{16}$$

Then, by substituting Eqn. 15, into Eqn. 16, Eqn 17 can be obtained.

$$S = \sum_{i=1}^{n} (y_i - (a + b \cdot x_i + c \cdot z_i + d \cdot x_i^2 + e \cdot z_i^2 + f \cdot x_i \cdot z_i))^2 \tag{17}$$

For a minimum S, the partial derivatives of S with respect to the model parameters should be 0. Thus, 6 equations with 6 unknowns are obtained:

$$\frac{dS}{da} = 0, \frac{dS}{db} = 0, \frac{dS}{dc} = 0, \frac{dS}{dd} = 0, \frac{dS}{de} = 0, \frac{dS}{df} = 0 \tag{18}$$

Arranging these equations as a matrix yields the following matrix:

$$\begin{bmatrix} N & S_x & S_z & S_{xx} & S_{zz} & S_{xz} \\ S_x & S_{xx} & S_{xz} & S_{xxx} & S_{xzz} & S_{xxz} \\ S_z & S_{xz} & S_{zz} & S_{xxz} & S_{zzz} & S_{xzz} \\ S_{xx} & S_{xxx} & S_{xxz} & S_{xxxx} & S_{xxzz} & S_{xxxz} \\ S_{zz} & S_{xzz} & S_{zzz} & S_{xxzz} & S_{zzzz} & S_{xzzz} \\ S_{xz} & S_{xxz} & S_{xzz} & S_{xxxz} & S_{xzzz} & S_{xxzz} \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = \begin{bmatrix} S_y \\ S_{xy} \\ S_{yz} \\ S_{xxy} \\ S_{yzz} \\ S_{xyz} \end{bmatrix} \tag{19}$$

where N is the total number of points and S stands for the summation of the terms. For example, S_{xy} is defined as

$$S_{xy} = \sum_{i=1}^{n} x_i \cdot y_i \tag{20}$$

Although some parts of the obstacle have a similar height to the road plane, in a regular road scene, a

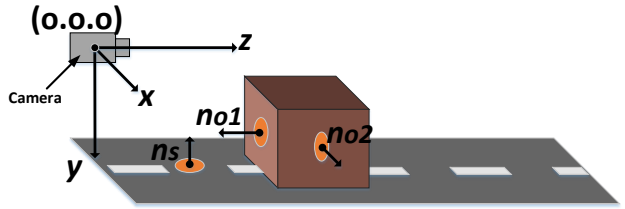

Fig. 4 A typical road scene: example planar patches are depicted by orange ellipses and the directions of their surface normals are illustrated by the arrows pointing out from them. It can be seen that the direction of a surface normal from the road (i.e. n_s) and the direction of a surface normal from the obstacle (i.e. n_{o1} and n_{o2}) are typically perpendicular to each other.

point on an obstacle has an almost perpendicular surface orientation to the road, as illustrated in Figure 4. Thus, to minimize the effect of the outliers, both the distances and the surface orientations of the points are minimized.

In [18], it is shown that the least squares line fitting can be improved by incorporating gradient orientations along with the point locations from the edges. In a road model estimation application, the selected road model is a $2D$ quadratic surface and this model needs to be fitted to the planar patches with known gradients.

For a given point i, the gradient of its planar patch and the gradient of the quadratic road surface model should be the same for the location of point $i([x_i, z_i])$. The parameters of the planar patches $[n_x, n_y, n_z]$ are already calculated in section 4 and the gradients of a patch can be estimated as $mg = \delta y/\delta x = -n_x/n_y$ and $ng = \delta y/\delta z = -n_z/n_y$. Then, the gradients of the road model (Mg, and Ng) can be calculated as in Eq. (21) and Eq. (22).

$$Mg = \frac{dY}{dx} = b + 2 \cdot d \cdot x + f \cdot z \tag{21}$$

$$Ng = \frac{dY}{dz} = c + 2 \cdot e \cdot z + f \cdot x \tag{22}$$

To minimize the gradient differences along with the height, Eq. (17) is modified as Eq. (23).

$$S = \sum_{i=1}^{n}(y_i - (Y_i))^2 + G \cdot [\sum_{i=1}^{n}(mg_i - (Mg_i))^2 + \sum_{i=1}^{n}(ng_i - (Ng_i))^2] \tag{23}$$

where G is a constant (i.e. 1). Replacing Y, M_g, and N_g with Eq. (15), Eq. (21), and Eq. (22) in Eq. (23) yields Eq. (24).

$$S = \sum_{i=1}^{n}(y_i - (a + b \cdot x_i + c \cdot z_i + d \cdot x_i^2 + e \cdot z_i^2 + f \cdot x_i \cdot z_i))^2 + G \cdot [\sum_{i=1}^{n}(mg_i - (b + 2 \cdot d \cdot x_i + f \cdot z_i))^2 + \sum_{i=1}^{n}(ng_i - (c + 2 \cdot e \cdot z_i + f \cdot x_i))^2] \tag{24}$$

For a minimum S, the partial derivatives of S with respect to the model parameters should be 0 (see equation Eq. (18)). Thus, 6 equations with 6 unknowns are obtained. Arranging these equations as a matrix yields Eq. (25) (on page 6). In equation Eq. (25) compared to equation Eq. (19) new terms are depicted in red.

Solving Eq. (25) yields all the parameters for the road surface model.

6 SURFACE FITTING WITH BI-SQUARE UPDATE

The least-squares fitting is a robust approach for curve fitting. However, as their residuals are squared, it is sensitive to extreme outliers. To minimize the effect of extreme outliers, fitting is applied iteratively with a bi-square update. However, in this paper along with the difference in height, the difference in patch-normals are also minimized (using Eq. (25)). Furthermore, weighting updates are estimated by considering both the difference in height and the difference in patch-normals. It can be simplified to the following steps, and iterated until it converges.

1. Fit the Euclidean road candidates to the road model using Eq. 25.

2. Calculate the bi-square weighting with Eq. (26) for each data point.

$$w_i = \frac{1}{\Delta y_i^2 + \Delta n_i^2 + \Delta m_i^2} \tag{26}$$

where, for point i, x_i, y_i and z_i are its known world coordinate system. n_i and m_i are already calculated from the 2D disparity map and Δy_i, Δn_i and Δm_i are defined as follows (the required parameters for Y, N_g and M_g are estimated in step 1):

$$\Delta y_i = y_i - Y(x_i, z_i) \tag{27}$$

$$\Delta n_i = ng_i - Ng(x_i, z_i) \tag{28}$$

$$\Delta m_i = mg_i - Mg(x_i, z_i) \tag{29}$$

3. Rerun step 1 with bi-square weightings applied to each term, thus reducing the effects of the outliers.

$$\begin{bmatrix} N & S_x & S_z & S_{xx} & S_{zz} & S_{xz} \\ S_x & S_{xx}+G\cdot N & S_{xz} & S_{xxx}+2\cdot G\cdot S_x & S_{xzz} & S_{xxz}+G\cdot S_z \\ S_z & S_{xz} & S_{zz}+G\cdot N & S_{xxz} & S_{zzz}+2\cdot G\cdot S_z & S_{xzz}+G\cdot S_x \\ S_{xx} & S_{xxx}+2\cdot G\cdot S_x & S_{xxz} & S_{xxxx}+4\cdot G\cdot S_{xx} & S_{xxzz} & S_{xxxz}+2\cdot G\cdot S_{xz} \\ S_{zz} & S_{xzz} & S_{zzz}+2\cdot G\cdot S_z & S_{xxzz} & S_{zzzz}+4\cdot G\cdot S_{zz} & S_{xzzz}+2\cdot G\cdot S_{xz} \\ S_{xz} & S_{xxz}+G\cdot S_z & S_{xzz}+G\cdot S_x & S_{xxxz}+2\cdot G\cdot S_{xz} & S_{xxzz}+2\cdot G\cdot S_{xz} & S_{xxzz}+G\cdot(S_{zz}+S_{xx}) \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \\ e \\ f \end{bmatrix} = \begin{bmatrix} S_y \\ S_{xy}+G.S_m \\ S_{yz}+G.S_n \\ S_{xxy}+2.G.S_{mx} \\ S_{yzz}+2G.S_{nz} \\ S_{xyz}+G.(S_{xz}+S_{nx}) \end{bmatrix} \tag{25}$$

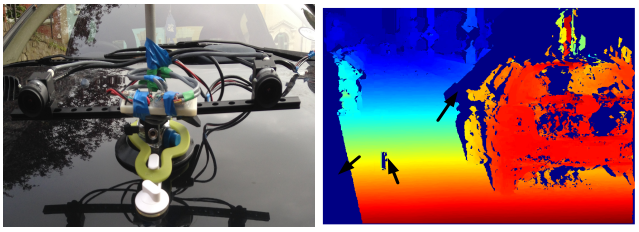

Fig. 5 Example experimental set up, where two point grey cameras synchronized by a PWM signal generated by an Arduino board is shown on the left, and example output disparity map with marked noise sources, where example error sources include saturation in the image and occluded areas is shown on the right.

7 EXPERIMENTAL RESULTS

7.1 Experimental set-up

The algorithm proposed in this paper uses a dense disparity map as input. For our stereovision camera rig, two point grey Flea3 (FL3-GE-13S2C-CS) cameras have been used and these cameras are synchronized by a PWM signal generated by an Arduino board. The baseline of the cameras is set to 34 cm. The constructed camera rig is, then, simply placed on the vehicle with air suction pads, without concern about any external camera parameters since the proposed algorithm does not need any input external camera parameters, for instance pitch, yaw or roll angle. An example hardware set-up and an example output disparity map are shown in Figure 5.

7.2 Quantitative results

In this paper, the road is modeled with a quadratic model, and the parameters of this model are estimated. The used road surface model is defined in Eqn. 1, where [a,b,c,d,e,f] are the parameters of the model. To quantify and compare this algorithm, ideally these parameters needs to be known. However, selecting these values manually is not possible, and the ground truth of these values are not available. For this reason, first, we have manually selected the ground truth road area, in a sequence of 1084 frames, in the image domain. Thus, it is known that, whether a point belongs to the road or not. Then, the road is labeled using the estimated road model. This is done by thresholding a pixel (i) based on its distance to the road model ($Y_i - y_i < T$), where Y_i is defined in Eqn, 15, and y_i is the height of the pixel and T is the threshold (i.e. 5 cm). Then, if the distance is smaller than the threshold, that point is labeled as a road point. Otherwise, the point is labeled as a non-road point. When the road model fits well to the road, most of the points which belong to the road would be labeled as a road point by the algorithm. Thus, we would like to estimate the ratio between the correctly labeled road points by the algorithm, and the total number of the ground truth road points. This ratio is called the True Positive Rate (TPR), which is also known as Recall. The equation of TPR is defined in Eqn. 30.

$$TPR = \frac{T_p}{Tp + F_n} \tag{30}$$

where True Positive (T_p) is the total number of correctly selected road points, and False Negative (F_n) is the total number of road points that are incorrectly selected as non-road points. Thus, $T_p + F_n$ is equal to total number of ground truth road points available.

TPR (using a range of thresholds, from 5cm to 50cm) is estimated for a total number of 1084 frames (the ground truth free space is manually labeled with the help of a script). The results of the proposed method in this paper, and the algorithm described in [11] are illustrated in Figure 6. From Figure 6, it can be seen that the proposed method can estimate the road model accurately, even if a low threshold is applied (i.e. for small obstacles).

The accuracy of the algorithm with respect to depth is limited with the stereo vision input. The theoretical depth accuracy δz against depth z attainable for the stereo vision input can be calculated using Eqn. 31 [6], where δd is the accuracy in disparity. In this work, in order to maintain accurate segmentation of the road, a depth threshold in cm from the modeled road to the point cloud is used. Due to the squared increase in the standard deviation from the source data, the theoretical upper bound of the maximum distance can be calculated using Eqn. 31. Although a dynamic threshold can be applied following the squared rule, allowing the inclusion of far-field data points, we are more interested in a consistently accurate fitting results throughout.

$$\delta z = \frac{z^2}{f \cdot b_s} \cdot \delta d \tag{31}$$

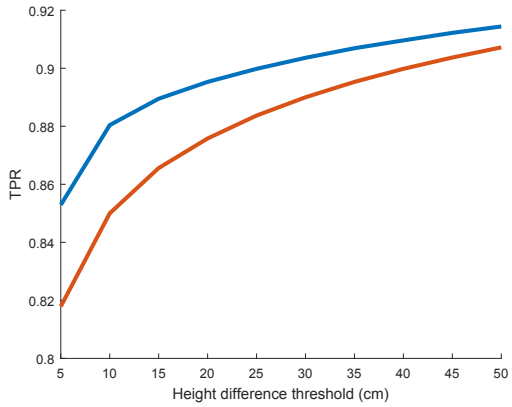

Fig. 6 Estimated TPR for the method described in this paper (shown with the blue line) and TPR for the algorithm described in [11] (shown with the red line).

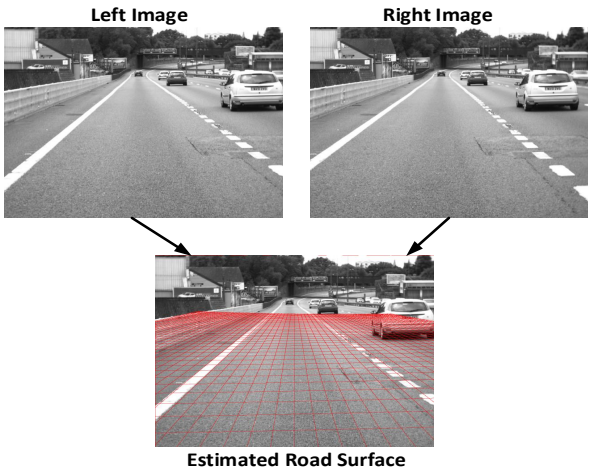

Fig. 7 The input and the output of the algorithm, where the algorithm uses stereo image pairs supplied by the stereo vision camera and estimates the road surface using a road model

In Figure 7, the input (left and right images from the stereo vision camera) and the output (estimated road surface model) of the algorithm are demonstrated. Sample experimental results are illustrated in Figure 8, where the estimated road area is highlighted with green. Also, a sample video sequence, where the input disparity map, segmented road area, and overlaid road model are illustrated, is publicly available at [1].

The proposed algorithm has been implemented in C. In Table 1, the detailed run time for each operation using a single thread on an Intel $i7-870$ CPU is shown, where the described algorithm is suitable for parallel processing. For instance, the test CPU can run 8 threads in parallel and ideally 8 times improvement in run time is expected. In future, GPU implementation of the algorithm is also planned.

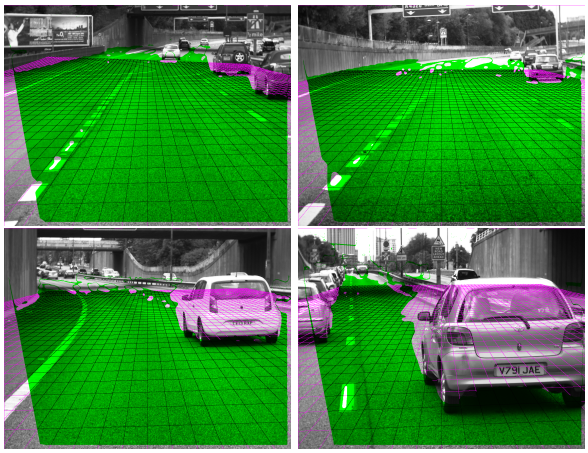

Fig. 8 Sample experimental results (segmented for $\pm 10cm$). In these images, pixels with an unknown disparity value such as for occluded areas (i.e. the left hand side of the cars) are also segmented as non-road areas.

Table 1 Run time of the operations

Operation	Run time
Disparity map to WCS conversion	7 ms
Surface normal estimation	26 ms
Quadratic fitting	13 ms
Weighted quadratic surface fitting	25 ms

8 CONCLUSION

In this paper, a novel road model estimation algorithm is described. An accurately, estimated road model, is important for many ADAS applications such as lane detection, obstacle detection and road marking recognition. The proposed algorithm utilizes the efficiently estimated planar patches and exploits patch orientations to minimize the effects of outliers in the road model estimation stage. The described algorithm integrates patch orientations to the fitting stage. Fitting has been performed iteratively and, in each iteration, based on the patch orientation and the road model orientation for a given point, a weight is calculated for all the points on the 3D map. Thus, the effect of outliers is minimized for accurate road model estimation. To demonstrate robustness of the system, the road is segmented based on an estimated road model (± 10 cm) and TPR is calculated as 88%. It should be noted that, the proposed method does not need any extrinsic camera parameters and the stereo camera-rig is simply plugged on to a vehicle without any concern about the extrinsic camera parameters. Future work includes tracking the road model parameters for more robust results.

Acknowledgements The authors would like to thank Brett Hosking for his valuable support during recording experimental sequences.

References

1. Example road segmentation sequence (2015). URL https://drive.google.com/file/d/0B-8QPnuuk-qdS09QTzdKdkxPOU0/view
2. Chapuis, R., Aufrere, R., Chausse, F.: Accurate road following and reconstruction by computer vision. Intelligent Transportation Systems, IEEE Transactions on **3**(4), 261–270 (2002)
3. Demirdjian, D., Darrell, T.: Motion estimation from disparity images. In: Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on, vol. 1, pp. 213–218. IEEE (2001)
4. Eidehall, A., Pohl, J., Gustafsson, F.: Joint road geometry estimation and vehicle tracking. Control Engineering Practice **15**(12), 1484–1494 (2007)
5. Einecke, N., Eggert, J.: A two-stage correlation method for stereoscopic depth estimation. In: DICTA, pp. 227–234 (2010)
6. Gallup, D., Frahm, J.M., Mordohai, P., Pollefeys, M.: Variable baseline/resolution stereo. In: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, pp. 1–8. IEEE (2008)
7. Gao, Y., Ai, X., Wang, Y., Rarity, J., Dahnoun, N.: Uv-disparity based obstacle detection with 3d camera and steerable filter. In: Intelligent Vehicles Symposium (IV), 2011 IEEE, pp. 957–962. IEEE (2011)
8. Garcia-Fernandez, A.F., Hammarstrand, L., Fatemi, M., Svensson, L.: Bayesian road estimation using onboard sensors. Intelligent Transportation Systems, IEEE Transactions on **15**(4), 1676–1689 (2014)
9. Geiger, A., Roser, M., Urtasun, R.: Efficient large-scale stereo matching. In: ACCV (2010)
10. Hu, Z., Lamosa, F., Uchimura, K.: A complete uv-disparity study for stereovision based 3d driving environment analysis. In: 3-D Digital Imaging and Modeling, 2005. 3DIM 2005. Fifth International Conference on, pp. 204–211. IEEE (2005)
11. Hu, Z., Uchimura, K.: Uv-disparity: an efficient algorithm for stereovision based scene analysis. In: Intelligent Vehicles Symposium, 2005. Proceedings. IEEE, pp. 48–54. IEEE (2005)
12. Labayrade, R., Aubert, D., Tarel, J.P.: Real time obstacle detection in stereovision on non flat road geometry through v-disparity representation. In: Intelligent Vehicle Symposium, 2002. IEEE, vol. 2, pp. 646–651. IEEE (2002)
13. Lazaros, N., Sirakoulis, G.C., Gasteratos, A.: Review of stereo vision algorithms: from software to hardware. International Journal of Optomechatronics **2**(4), 435–462 (2008)
14. Lundquist, C., Schön, T.B.: Joint ego-motion and road geometry estimation. Information Fusion **12**(4), 253–263 (2011)
15. Manduchi, R., Castano, A., Talukder, A., Matthies, L.: Obstacle detection and terrain classification for autonomous off-road navigation. Autonomous robots **18**(1), 81–102 (2005)
16. Nedevschi, S., Schmidt, R., Graf, T., Danescu, R., Frentiu, D., Marita, T., Oniga, F., Pocol, C.: 3d lane detection system based on stereovision. In: Intelligent Transportation Systems, 2004. Proceedings. The 7th International IEEE Conference on, pp. 161–166. IEEE (2004)
17. Oniga, F., Nedevschi, S.: Processing dense stereo data using elevation maps: Road surface, traffic isle, and obstacle detection. Vehicular Technology, IEEE Transactions on **59**(3), 1172–1182 (2010)
18. Petkovic, T., Loncaric, S.: Using gradient orientation to improve least squares line fitting. In: Computer and Robot Vision (CRV), 2014 Canadian Conference on, pp. 226–231. IEEE (2014)
19. Sappa, A.D., Dornaika, F., Ponsa, D., Gerónimo, D., López, A.: An efficient approach to onboard stereo vision system pose estimation. Intelligent Transportation Systems, IEEE Transactions on **9**(3), 476–490 (2008)
20. Sappa, A.D., Herrero, R., Dornaika, F., Gerónimo, D., López, A.: Road approximation in euclidean and v-disparity space: a comparative study. In: Computer Aided Systems Theory–EUROCAST 2007, pp. 1105–1112. Springer (2007)
21. Viejo, D., Garcia, J., Cazorla, M., Gil, D., Johnsson, M.: Using gng to improve 3d feature extractionpplication to 6dof egomotion. Neural Networks **32**, 138–146 (2012)
22. Wang, Y., Dahnoun, N., Achim, A.: A novel system for robust lane detection and tracking. Signal Processing **92**(2), 319–334 (2012)
23. Wedel, A., Badino, H., Rabe, C., Loose, H., Franke, U., Cremers, D.: B-spline modeling of road surfaces with an application to free-space estimation. Intelligent Transportation Systems, IEEE Transactions on **10**(4), 572–583 (2009)
24. Zhang, Z.: A stereovision system for a planetary rover: Calibration, correlation, registration, and fusion. Machine Vision and Applications **10**(1), 27–34 (1997)
25. Zhang, Z., Ai, X., Dahnoun, N.: Efficient disparity calculation based on stereo vision with ground obstacle assumption. In: Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 21st European, pp. 1–5. IEEE (2013)
26. Zhang, Z., Ai, X., john, R., Dahnoun, N.: Obstacle detection using u-disparity on quadratic road surfaces. In: Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on, pp. 1352 – 1357. IEEE (2013)